GPBR Exercise 2

Name: Soi Zhi Wen

Matriculation Number: 22-132-245

1. In the context of fingerprint image analysis, exact graph matching paradigms pose significant challenges. In the example given, each node in the graph represents a unique location in the fingerprint image, and is labeled with a two-dimensional attribute that gives its position in the plane. The edges between nodes represent the ridges in the fingerprint image, which may vary in shape and orientation.

   Exact graph matching algorithms aim to find a one-to-one correspondence between the nodes and edges of two graphs such that the dissimilarity score between the two graphs is minimized. However, in the case of fingerprint image analysis, even a small variation in the position or shape of the ridges can lead to a significant difference in the graph representation. This means that exact graph matching algorithms may not be able to accurately match the ridges in two fingerprint images, leading to a high dissimilarity score.

   On the other hand, error-tolerant graph matching algorithms can be used to overcome these challenges. These algorithms allow for some level of variation in the position and shape of the ridges, and aim to find a correspondence between the nodes and edges of two graphs that minimizes the dissimilarity score while also accounting for these variations. In the case of fingerprint image analysis, error-tolerant graph matching algorithms can be used to find a correspondence between the ridges in two fingerprint images that takes into account variations in the shape and position of the ridges.

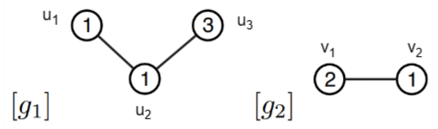
   In conclusion, for the given application of fingerprint image analysis, an error-tolerant graph matching paradigm would be more appropriate than an exact graph matching paradigm. This is because the exact graph matching paradigm would not be able to accurately match the ridges in the fingerprint images, whereas error-tolerant graph matching algorithms can handle variations in the ridge structure while still finding a good correspondence between the two graphs.

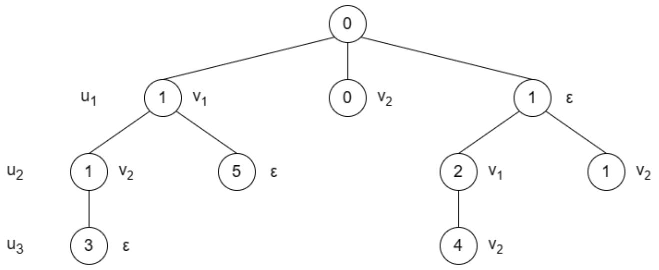
2. Node operations:
   - Node insertion: We can assign a cost of infinity for node insertion, as adding a node can never make a graph isomorphic to a subgraph of another graph.
   - Node deletion: We can assign a cost of infinity for node deletion, as removing a node can never make a graph isomorphic to a subgraph of another graph.
   - Node substitution: We can assign a cost of 1 for node substitution, as replacing a node with another node that has the same attributes does not affect the subgraph isomorphism.

Edge operations:

- Edge insertion: We can assign a cost of infinity for edge insertion, as adding an edge can never make a graph isomorphic to a subgraph of another graph.
- Edge deletion: We can assign a cost of infinity for edge deletion, as removing an edge can never make a graph isomorphic to a subgraph of another graph.
- Edge substitution: We can assign a cost of 1 for edge substitution, as replacing an edge with another edge that connects the same nodes does not affect the subgraph isomorphism.

3.





4. From the scatter plots, it can be observed that there is a strong positive correlation between the exact and approximate graph edit distances computed using BP-GED. This means that as the exact graph edit distance increases, the approximate graph edit distance computed using BP-GED also tends to increase.

   The scatter plot for Graph A (AIDS) shows that the approximate graph edit distance computed using BP-GED is generally higher than the exact graph edit distance. This means that BP-GED tends to overestimate the graph edit distance on this dataset. However, the correlation between the exact and approximate graph edit distances is strong, indicating that BP-GED can still be a useful approximation method.

   On the other hand, the scatter plot for Graph B (FP) shows a much stronger correlation between the exact and approximate graph edit distances computed using BP-GED. Additionally, the results of $d'\psi$ are generally closer to the exact graph edit distance compared to the results of $d\psi$. This suggests that BP-GED is a more accurate approximation method for this dataset compared to Graph A (AIDS).

   Comparing the results obtained using BP-GED with those obtained using GED, it can be seen that BP-GED tends to overestimate the graph edit distance on Graph A (AIDS), while it is a more accurate approximation method for Graph B (FP). However, BP-GED is generally faster and more efficient compared to GED, making it a useful method for large-scale graph edit distance computations.